PCT

WORLD INTELLECTUAL PROPERTY ORGANIZATION
International Bureau

INTERNATIONAL APPLICATION PUBLISHED UNDER THE PATENT COOPERATION TREATY (PCT)

(51) International Patent Classification 6 :
H04J 3/16, 3/26, H04Q 11/04

A1

(11) International Publication Number: WO 96/19882

(43) International Publication Date: 27 June 1996 (27.06.96)

(21) International Application Number: PCT/US95/14637

(22) International Filing Date: 26 October 1995 (26.10.95)

(30) Priority Data:
08/358,427 19 December 1994 (19.12.94) US

(71) Applicant: MOTOROLA INC. [US/US]; 1303 East Algonquin Road, Schaumburg, IL 60196 (US).

(72) Inventors: YONG, Mei; 3 Ledgewood Drive, Canton, MA 02021 (US). YANG, Jian; Apartment #102, 350 Engamore Lane, Norwood, MA 02062 (US). NG, Dennis; 126 Indian Meadow Drive, Northboro, MA 01532 (US).

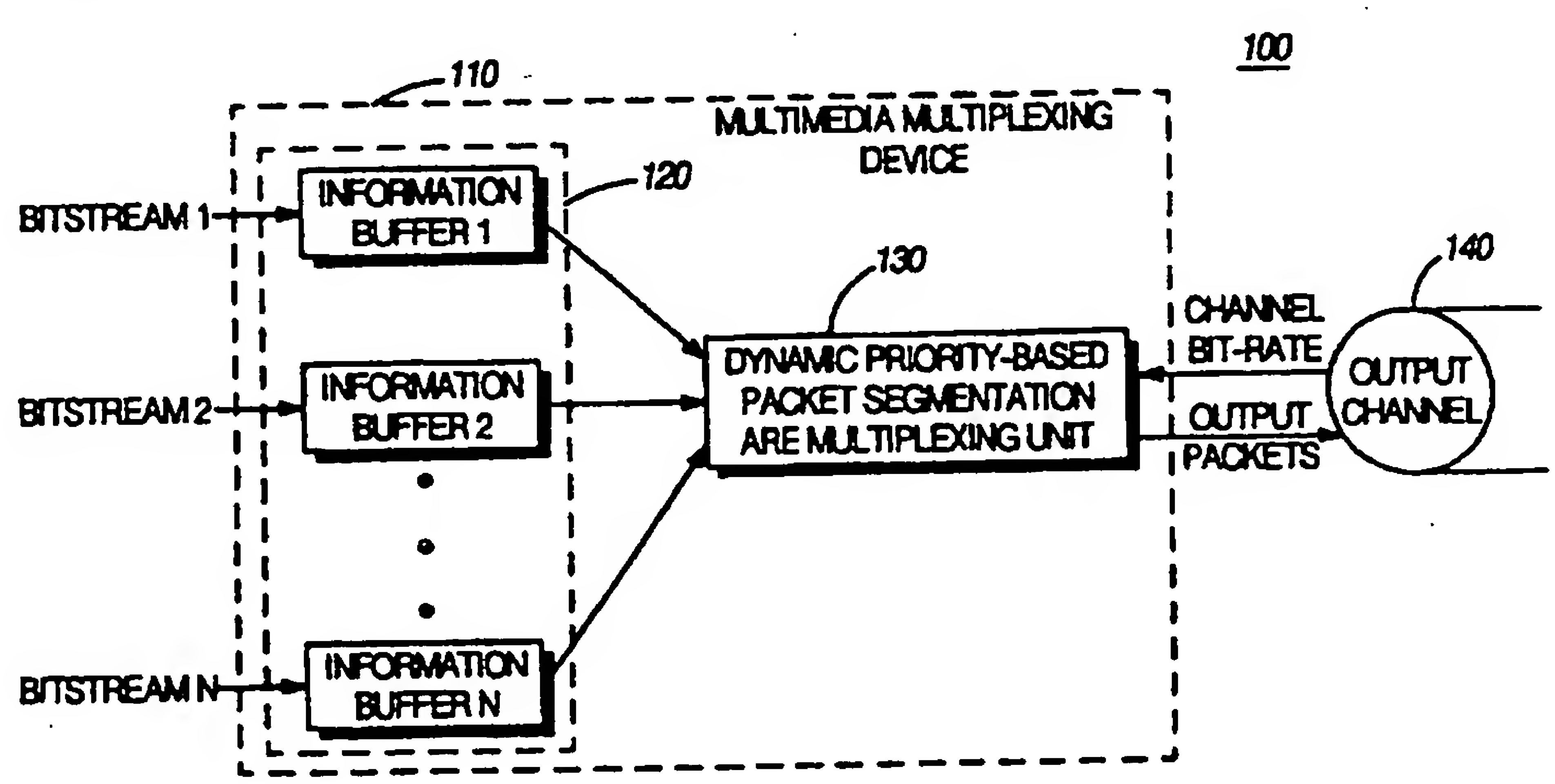
(74) Agents: STOCKLEY, Darleen, J. et al.; Motorola Inc., Intellectual Property Dept., 1303 East Algonquin Road, Schaumburg, IL 60196 (US).

(81) Designated States: CA, CN, European patent (AT, BE, CH, DE, DK, ES, FR, GB, GR, IE, IT, LU, MC, NL, PT, SE).

Published
With international search report.

BEST AVAILABLE COPY

(54) Title: MULTIMEDIA MULTIPLEXING DEVICE AND METHOD USING DYNAMIC PACKET SEGMENTATION



(57) Abstract

The present invention provides a method (900) and device (100) in multimedia communication systems for efficiently segmenting information bitstreams from multiple media sources into variable length packets, and multiplexing and sending the packets to a shared communication link with low delay and low overhead. The packet segmentation and multiplexing are performed dynamically based on fullness of a set of information buffers that contain the information bitstreams to be transmitted, and delay-sensitivity of each information bitstream. The multi-discipline queuing scheme has been developed in this invention to control the dynamic packet segmenting and multiplexing process.

FOR THE PURPOSES OF INFORMATION ONLY

Codes used to identify States party to the PCT on the front pages of pamphlets publishing international applications under the PCT.

AT	Austria
AU	Australia
BB	Barbados
BE	Belgium
BF	Burkina Faso
BG	Bulgaria
BJ	Benin
BR	Brazil
BY	Belarus
CA	Canada
CF	Central African Republic
CG	Congo
CH	Switzerland
CI	Côte d'Ivoire
CM	Cameroon
CN	China
CS	Czechoslovakia
CZ	Czech Republic
DE	Germany
DK	Denmark
ES	Spain
FI	Finland
FR	France
GA	Gabon
GB	United Kingdom
GE	Georgia
GN	Guinea
GR	Greece
HU	Hungary
IE	Ireland
IT	Italy
JP	Japan
KE	Kenya
KG	Kyrgystan
KP	Democratic People's Republic of Korea
KR	Republic of Korea
KZ	Kazakhstan
LI	Liechtenstein
LK	Sri Lanka
LU	Luxembourg
LV	Latvia
MC	Monaco
MD	Republic of Moldova
MG	Madagascar
ML	Mali
MN	Mongolia
MR	Mauritania
MW	Malawi
NE	Niger
NL	Netherlands
NO	Norway
NZ	New Zealand
PL	Poland
PT	Portugal
RO	Romania
RU	Russian Federation
SD	Sudan
SE	Sweden
SI	Slovenia
SK	Slovakia
SN	Senegal
TD	Chad
TG	Togo
TJ	Tajikistan
TT	Trinidad and Tobago
UA	Ukraine
US	United States of America
UZ	Uzbekistan
VN	Viet Nam

MULTIMEDIA MULTIPLEXING DEVICE AND METHOD USING DYNAMIC PACKET SEGMENTATION

Field of Invention

This invention relates generally to multimedia communications and more particularly to multimedia multiplexing.

Background

Recent advances in telecommunications and Digital Signal Processing (DSP), technologies, have created a growing demand for multimedia communication products for both business and home use. Multimedia communications often involve the simultaneous transmission of audio, video and data, such as graphics, fax or computer data, through an available shared communication link. To make efficient use of the available communication link, a number of techniques are required. For example, compression algorithms for compressing various media types are needed to reduce the bandwidth needed to transmit them. In addition, an efficient and flexible multiplexing method is needed to provide an acceptable quality of service, i.e., low multiplexing overhead and queuing delay, for each media type.

In traditional circuit-switched networks, different signals are multiplexed together using time division multiplexing (TDM). In TDM, a fixed bandwidth is typically allocated to each media for the duration of a call, and there is little flexibility to take advantage of the bursty nature of data, video and audio information.

To gain more flexibility and efficiency, packet multiplexing techniques have been proposed. Packet multiplexing has been widely adopted in ATM networks. In packet multiplexing, each information bitstream is segmented into packets, and packets from different bitstreams are multiplexed and sequentially transmitted over a communication channel. Each packet typically contains a header field and a payload field. A packet header contains a packet identifier which is used for recovering each individual information bitstream from a multiplexed packet sequence. The payload field of a packet may optionally contain some media-specific adaptation information in addition to actual information bits. Packets may be of fixed-length or of variable-length. Fixed-length packets used in ATM have the following advantages: fast segmentation and reassembly, no need for delineation flags, and easy synchronization. However, fixed-size packets are not suitable for use on low-speed links, because of efficiency and delay considerations. Variable-length packets simplify the implementation of the adaptation layers and allow a flexible design to trade-off delay against efficiency. Thus, variable-length packets are more appealing for use on low-speed links such as voiceband modem links.

When variable-length packets are used, flags are inserted between packets for delineation and synchronization. A widely used variable-length packet format is the HDLC-based framing structure, where HDLC represents High-level Data Link Control. In the HDLC format, the packet delineation flag, called the HDLC flag, is a one-byte binary word: "01111110". To avoid duplication of the HDLC flag in the information bitstream, HDLC bit stuffing is applied to the content of a packet between two flags by inserting a "0" bit after every five contiguous "1" bits. Overhead

caused by HDLC bit-stuffing of a random bitstream is approximately 1.6%, but can be as high as 20% in the worst case. Another use of the HDLC flag is in bit-rate adaptation. When the aggregate bit rate is less than the channel rate, the HDLC flag can be repetitively sent during channel idle periods.

The effectiveness of a packet multiplexing scheme typically depends on its efficiency and delay. Efficiency is obtained by reducing packet overhead and by maximizing the bandwidth utilization. Increasing the packet size reduces the effective packet overhead, but increases the queuing delay. Thus, there is a need for a device and method that can achieve a good trade-off between efficiency and delay in a multimedia communication system .

Description of the Drawings

FIG. 1 is a block diagram showing one embodiment of a multimedia multiplexing device in accordance with the present invention.

FIG. 2 is a block diagram showing the multimedia multiplexing device of FIG. 1 with greater particularity.

FIG. 3 is a flow chart showing the processes and control flow for multiplexing different priority groups according to the Head of Line Priority for Variable Length Packet (HOLP-VLP) queuing discipline.

FIG. 4 is a flow chart showing the processes and control flow for multiplexing different bitstreams in the same priority

group according to the Weighted Round Robin for Variable Length Packet (WRR-VLP) queuing discipline.

FIG. 5 is shows an example of one embodiment of steps wherein a hybrid queuing discipline is applied to multimedia bitstreams in accordance with the present invention.

FIG. 6 is a flow chart illustrating steps that describe the high-level processes and control flow in the multimedia multiplexing device in accordance with the present invention.

FIG. 7 is a diagram showing pre-HDLC packets, and the interrupts associated with those packets after HDLC encoding.

FIG. 8 is a block diagram showing one embodiment of Data Communications Equipment/Data Terminal Equipment having a multimedia multiplexing device in accordance with the present invention.

FIG. 9 is a flow chart showing one embodiment of steps in accordance with the method of the present invention .

Detailed Description of the Invention

The present invention provides a device and method for a multimedia communication system in which multiple information bitstreams are prioritized and dynamically segmented into variable-length packets, and multiplexed for efficient transmission over a digital communication link.

Page missing upon filing

THIS PAGE BLANK (USPTO)

information buffer and passing the selected bitstream to a packet generator (250); and a packet generator (250), that is operably coupled to receive a bitstream from the server (240), for forming packets, sending packets to an output channel, and informing the queuing and segmentation controller (230) each time a packet is sent. Where selected, flags and stuffing bits for a packet may be generated by a hardware HDLC controller (260) that is operably coupled to the packet generator (250).

There are two main aspects of the device and method of the present invention: A) it dynamically adjusts packet sizes based on the fullness of the information buffers and available bit-rate of the output channel; B) it multiplexes packets in an order based on a predetermined queuing discipline which gives higher priority to delay-sensitive source(s) and allows effective bandwidth sharing among different sources. In the following, the detailed description of the algorithm and an implementation in accordance with this invention are provided.

First, bitstreams from different sources are prioritized based on their delay tolerances, where a bitstream that can tolerate the least delay, for example, real-time traffic, is given the highest priority, a bitstream that can tolerate the most delay, for example, non-real-time traffic, is given the lowest priority, and bitstreams that have an equal delay tolerance are given the same priority. Furthermore, the bitstreams having an equal priority are grouped into a single priority group. Each priority group may contain one or more bitstreams.

A bandwidth-weighting factor for bitstream i, denoted as a_i, is defined as a fraction of bandwidth allocated to the bitstream i out of the total bandwidth allocated to the priority

group containing bitstream i. If there is only one source in a priority group, then $a_i = 1$.

Two often used queuing disciplines, known in the art, for packet multiplexing are called Head-Of-Line-Priority (HOLP) and Weighted-Round-Robin (WRR). Where there are N buffers, represented as B_1, B_2,..., and B_N, for storing source bitstream 1, 2,..., and N respectively, the priority of each buffer is the same as the priority of the corresponding bitstream. Also, priorities of buffers are P_1, P_2,..., and P_N, with $P_1 > P_2 > ..., P_N$. These buffers are served (where serving a buffer means taking bits out of a buffer and sending them to an output port) according to a predetermined queuing discipline. Where the predetermined queuing discipline is HOLP, every time the server finds the channel ready to accept a packet, it examines B_1 first, B_2 next and so on until it finds a packet. Where the predetermined queuing discipline is WRR, the server serves the buffers cyclically in a predetermined order. In any such repetitive cycle, it examines each buffer a specified number of times in proportion to its weight.

There is a key difference between the present invention and multiplexing using a HOLP or WRR queuing discipline. In the HOLP or WRR queuing discipline, the priority and bandwidth allocation to different information sources are guaranteed by serving each specified buffer either more or less frequently, but there is no improvement to packetization efficiency. Using the present invention, priority and bandwidth allocation are achieved by dynamically adjusting packet size. This method not only achieves efficient bandwidth sharing and guarantees low delay to high priority bitstreams, but also optimizes packet size to minimize packetization overhead.

In the present invention, two different queuing disciplines are defined and used for variable length packet (VLP) segmentation, referred as the HOLP-VLP and the WRR-VLP.

The HOLP-VLP queuing discipline of the present invention is used to treat sources or priority groups which all have different priorities. The HOLP-VLP queuing discipline includes the steps described below. Where buffer i, i an index of a buffer, is currently being served, a server continues to serve buffer i until one of the following two events happens: a) at least one packet from a higher priority buffer is ready to be sent; b) there are not enough bits in buffer i to form a packet. When either of the above two events happens, the server stops serving the current buffer as soon as possible and then switches to serve the next buffer which has the highest priority among all the buffers that have packets ready to be sent, where said "as soon as possible" means that the server needs to complete sending the packet currently being sent before making a switch. The length of a packet from a buffer is upper-bounded by a maximum number of bits which can be sent during the time when said buffer is being continuously served. The detailed determination of a packet length is given in EQ.1 and EQ.2 below.

FIG. 3, numeral 300, is a flow chart showing the processes and control flow for multiplexing M, where M is an integer greater than 1, different priority groups according to the HOLP-VLP queuing discipline. In FIG. 3, PG_i is the an abbreviation for Priority Group i for i = 1, 2,..., M, and the priority of PG_1 is higher than that of PG_2, the priority of PG_2 is higher than that of PG_3, and so on. In this process, packets currently in PG_1 (310) are sent first, packets in PG_2 (320) are then sent if no packet in PG_1 is

ready to be sent, and so on. If there is no packet in all PGs, the process of sending padding packet(s) (360) is executed until said process is interrupted by a packet-transmission request, where any PG sends a packet-transmission request as soon as it has a packet ready to be sent. Any process of sending a particular PG can be interrupted by a packet-transmission request of any higher priority PG, but not by a lower priority PG. Where such an interrupt occurs, the current process is terminated promptly, and then the next process, determined by the positions of a plurality of switches (365, 370, 375,..., 380), is started. Where PG_1 sends a packet-transmission request, switch 2 (365) switches to location 1; otherwise, switch 2 moves to location 0. Switch i, where i > 2, functions as follows: where at least one of the switches above it (i.e., switch 2,..., switch i-1) is either in position 1 or position 2, switch i moves to location 1; otherwise where PG_{i-1} sends a packet-transmission request, switch i moves to location 2; otherwise switch i moves to location 0. Where a switch is positioned at 1 for PG_2, an interrupt of PG_2, an interrupt of PG_2 immediately starts processing of PG_1. Where a switch is positioned at 1 for PG_i, i>2, an interrupt of PG_i immediately starts processing PG_{n-1}, where n an integer less than i, and is determined by positions of switches 0 to i-1. Where a switch i is set to 2, PG_{i-1} is processed. Where a switch is set to 1, an interrupt of PG_i starts a next process PG_{n-1}, n an integer less than i, where switch n is set to 2. Where switch i is set to 0 for PG_i, no prior $PG_{1->i-1}$ is processed.

The WRR-VLP queuing discipline of the present invention is used to treat multiple sources within the same priority group. The WRR-VLP queuing discipline functions as follows: In a pre-selected partition period, referred as T_p, the m serves each buffer in the same priority group cyclically in a predetermined

order and for a period, referred as T_i (where i is the index of the buffer). The period T_i is determined as the following. Normally, T_i is taken to be $a_i{}^*T_p$, where a_i is the bandwidth-weighting factor used for source i. However, T_i may be shortened if the buffer i has no more bits to be sent or if a packet from a higher priority buffer is ready to be sent, or it can be extended if buffer i still has bits to be sent, but all other buffers in the same priority group have no bits to be sent. The upper bound of the partition period T_p is determined by the maximum queuing delay requirement for each bitstream in this priority group and the lower bound of T_p is determined by the packetization efficiency requirement. T_p may also be adjusted dynamically. If a particular buffer is allocated a period of $a_i{}^*Tp$ to send a packet, but this packet is interrupted by a packet-transmission request from a higher priority buffer, then the remaining credit is typically given to the same buffer when this priority group is served again in the following time. The length of a packet from a buffer is upper bounded by the maximum number of bits which can be sent during the time when said buffer is being continuously served. The detailed determination of a packet length is given in EQ.1 and EQ.3 of section 4.4.

FIG. 4, numeral 400, is a flow chart showing the processes and control flow for multiplexing bitstreams within a priority group according to the WRR-VLP queuing discipline. In FIG. 4, a set of individual processes are connected as a ring. These processes are executed cyclically in a predetermined order, as shown by the arrows. For example, the process may include:

sending Bitstream 2 (402) followed by:

interruption and Exit (404), or

sending Bitstream 3 (406),

followed by:

interruption and Exit (404), or
sending Bitstream i-n, n a positive integer,

...

followed by:
interruption and Exit (404), or
sending Bitstream i-1 (408),
followed by:
interruption and Exit (404), or
sending Bitstream i (410),
followed by:
interruption and Exit (404), or
sending Bitstream i + 1 (412),

...

followed by:
interruption and Exit (404), or
sending Bitstream 1 (414),
followed by:
interruption and Exit (404), or
return to sending Bitstream 2 (402) and
continuing as described above.

At the beginning of a communication, the first process to start with can be any one of the processes as set forth above, e.g., the process that has a packet ready first. After a first process is started, each time when the priority group of the first process is being processed, the first bitstream to be sent depends on where the previous execution of this priority group exits. Any of these individual processes can be terminated if a packet-transmission request is received from a higher priority group or if there are no more packets ready to be sent in this priority group. When a process is interrupted or there are no more

Page missing upon filing

THIS PAGE BLANK (USPTO)

Page missing upon filing

THIS PAGE BLANK (USPTO)

$$B_i(t,t_{itp}) = \min\{\max\{a_i{}^*R^*T_p, R^*(t_e - t)\}, R^*(t_h - t)\} - O_i(t) \quad \text{(EQ 3)}$$

where a_i is the bandwidth weighting factor for bitstream i, T_p the partition interval as described previously, t_e the time at which at least one of the buffers, which has equal priority to buffer i, has a packet ready to be sent, and R, t_h and $O_i(t)$ are the same as those defined for EQ.2. The first term $a_i{}^*R^*T_p$ in EQ. 3 is the bandwidth allocated to source i during the partition interval T_p. Inclusion of the second term $R^*(t_e - t)$ in EQ. 3 allows the current packet size to be extended beyond what has been allocated to it until at least one of the other buffers, which has equal priority to buffer i, has a packet to be sent. Inclusion of the third term $R^*(t_h - t)$ in EQ. 3 allows transmission of the current packet to be interrupted by a packet-transmission request from a higher priority buffer, where a buffer sends out its packet-transmission request as soon as it stores one packet worth of bits.

In some applications, a packet must contain an integer number of bytes, so the size $S_i(t)$ is selected to ensure that the total packet length contains an integer number of bytes.

Where none of the buffers has a packet ready to be sent, i.e., $S_i(t) = 0$ for all i, a form of padding packets such as HDLC flags, which may also be of an integer number bytes, are typically be sent to the channel.

FIG. 6, numeral 600, is a flow chart describing the high-level processes and control flow implemented in the multimedia multiplexing device in accordance with the present invention. In

FIG. 6, the configuration process (610) performs priority assignment, bandwidth allocation, and selection of a queuing discipline based on the priority assignment as described above. This process is typically executed at the beginning of a communication upon receipt of bitstreams, but it can also be initiated during a communication if there is any change in either channel condition or information sources. The buffer monitoring process (620) monitors a set of information buffers which receive the input source bitstreams, and generates a packet-transmission request for any buffer which has at least one packet's worth of bits ready to be sent. The queuing execution process (630) selects and segments bitstreams to be sent at different times based on a queuing discipline determined by the configuration process (610), packet-transmission requests determined by the buffer monitoring process (620), and packet sending indication determined by the packet generation process (640). Finally, the packet generation process (640) performs packetization by forming packets based on a selected protocol and inserting stuffing bits into packets, and then output packets to an output channel.

The buffer monitoring process (620) is more efficiently accomplished by monitoring at small, regular, predetermined intervals. Using small versus large intervals decreases the potential delay of a high priority packet, but increases the processing burden on the processor.

The frequency at which the buffer monitoring (620) is called may be reduced, and hence the processing burden, by scheduling a calling time based on the size of a data block being currently sent if it is known.

In the following example, an implementation based on HDLC framing structure and three different media types are used:

audio - having priority P_1
video - having priority P_2
data - having priority P_3

where $P_1 > P_2 > P_3$. Since audio has the highest priority, the goal is to preempt all others when an audio packet is ready to be sent. The remaining bandwidth is first allocated to video, and then finally to data.

The process begins when the first encoded audio frame is ready to be sent. The following steps of operation are then conducted:

A) An audio packet is formed based on a selected protocol, HDLC encoded, and then is output to the output channel.

B) The processor is programmed to schedule the next instance of the buffer monitoring (620), just prior to the current packet being completely transmitted to the output channel. One method is to use a timer interrupt, which is typically available in microprocessors and digital signal processors.

C) When the next instance of (620) occurs, the available bandwidth is computed using EQ. 2, where t_h is predictable because audio is typically formed at a deterministic periodic rate. At this instance, a video packet (or data packet if no video is available, or HDLC flags is neither is available) is formed based on a selected protocol, HDLC encoded and then is output, where the packet size is determined by EQ. 1.

D) If a next audio frame is still not ready, execute steps B) and C) again; otherwise go to step E).

E) repeat step A) to D).

With the above approach, since the buffer monitoring (620) is called just as the audio encoder has a packet available, audio does not incur any delay. In practice, the audio encoder may experience processing jitter because it may take different times to encode different audio frames. Thus, the audio encoder does not have an audio packet ready at each expected time. To alleviate this problem, the audio encoder may delay all its encoded audio frames by the maximum expected jitter prior to forwarding them to the multiplexer. This ensures that an audio packet is always avaiable at time K*th, where K is an integer, assuming that silence suppression is not used. In this simplified approach, video will not be able to immediately interrupt a data packet if the data packet was being sent prior to a video packet being ready. However, the maximum data packet size may be selected to be reasonably small so that video will not be blocked longer than a predetermined time.

The overhead Oi(t) given in EQ. 2 and EQ. 3 typically includes a packet header, stuffing bits and flags, where the number of bits used for the header is typically fixed. When HDLC framing is used to delineate packets, a single-byte binary word of "01111110" is used for the delineation flag. To avoid duplication of this HDLC flag in the information bitstream, HDLC bit-stuffing is applied to the content of each packet by inserting a "0' bit following every five consecutive "1" bits. The overhead caused by the bit-stuffing is approximately 1.6% for a random sequence, and up to

20% if the raw data sequence contains all "1's". If this overhead is not appropriately accounted for, and adjusted from the available bandwidth ($O_i(t)$ in EQ. 2 and EQ. 3), all traffic will eventually experience cummulative delay. However, by accurately estimating the amount of HDLC overhead and reducing the available bandwidth (i.e packet size), of the non-real-time traffic, the delay-sensitive real-time traffic will not be unnecessarily delayed.

For software implementations of HDLC bit-stuffing, the amount of stuffed bits is readily available to the queuing execution process, and hence the overhead $O_i(t)$ is easily determined. However, when a hardware HDLC controller is used, the bit-stuffing process is concealed from the user, so the amount of stuffed bits is not directly available, and must be estimated.

The flags and stuffing bits for a packet are generated by a hardware HDLC controller, and the available bandwidth $B_i(t,t_{itp})$ is estimated as the channel bit-rate multiplied by the buffer serving time, $t-t_{itp}$, and then is subtracted by the overhead corrections due to an underestimate of overhead of packets sent previously, where a correction for a packet is computed as $T*R-S$, where: A) T represents a time difference between an acknowledgment to the packet and that of an immediately preceeding packet, where the acknowledgment is typically generated by the HDLC controller upon completing transmission of each packet; B) R represents the channel bit-rate; and C) S represents the number of known information bits in the packet, such as the raw information bit, flags and packet header.

Present hardware HDLC controllers are capable of providing an interrupt whenever a buffer has been entirely HDLC encoded and sent to the transmit FIFO, as well as being able to specify insertion of an HDLC flag to terminate a packet. These features can be utilized to estimate the bit-stuffing overhead for that buffer. FIG. 7, numeral 700, shows an example of raw, i.e., pre-HDLC, packets being formed (701), (702), (703), (704), and then HDLC encoded. After each packet has been HDLC encoded, an interrupt is generated by the hardware controller (711), (712), (713), (714). The time interval between any two consecutive interrupts, multiplied by the output bit rate, equals the total number of HDLC encoded bits transmitted in the interval between the interrupts. This computation includes bit-stuffing, and possibly a flag if the buffer contained an end-of-packet. Since the size of the corresponding pre-HDLC encoded packet is known to the queuing execution process, the difference between the total number of HDLC encoded bits transmitted and the computed number of transmitted bits is the HDLC overhead. For example,

$$\text{overhead for } 702 = (t_{712} - t_{711}) * R - (\text{size of } 702) + \text{remaining fractional overhead}$$

where t_{711}, t_{712} are the times at which the interrupts 711 and 712 occurs, respectively. In practice, there will be fractional parts resulting from each overhead computation. These must be continuously accumulated until the error surpasses an integer number of bits, and then compensated.

Depending on the latencies within the system, when the overhead for (702) is computed, the packet that is being formed may be (703), (704), or yet a later packet. In most implementations, the latency can typically be one packet.

Shorter latency means that any HDLC overhead introduced in a packet can be quickly compensated by shortening the nontime-critical packets, so that delay for time-critical traffic is minimized. The packet size of the traffic is always selected as according to EQ. 1, 2, or 3.

The overhead estimate is recomputed for every block of data that is to be HDLC encoded and is used to adjust the size of the next packet to be multiplexed. Where the overhead is underestimated, then nontime-critical packets are larger than necessary, resulting in time-critical traffic incurring more delay. However, where the overhead is overestimated, then nontime-critical packets are smaller than necessary, resulting in more filler flags and reducing efficiency.

Clearly, the present invention may be implemented in numerous communication system devices. FIG. 8, numeral 800, is a block diagram showing one embodiment of Data Communications Equipment/Data Terminal Equipment (DCE/DTE) (802) having a multimedia multiplexing device in accordance with the present invention.

In addition, the method of the present invention may, for example, be embodied as shown in FIG. 9, numeral 900. The method includes the steps of: A) receiving bitstreams from different media sources and temporarily buffering the bitstreams in a plurality of information buffers (902); and B) utilizing (904) a dynamic priority-based packet segmentation and multiplexing unit having a multi-discipline queuing scheme for: B1) dynamically adjusting sizes of packets for the information bitstreams based on a fullness of each of the plurality of information buffers and an available bit-rate of the output

channel and B2) multiplexing the packets in an order based on the multi-discipline queuing scheme that assigns a higher priority to delay-sensitive source(s) than to delay-insensitive source(s) and provides an effective bandwidth sharing among a plurality of sources. Further implementation of the method proceeds as described above.

Although exemplary embodiments are described above, it will be obvious to those skilled in the art that many alterations and modifications may be made without departing from the invention. Accordingly, it is intended that all such alterations and modifications be included within the spirit and scope of the invention as defined in the appended claims.

We claim:

1. A multimedia multiplexing device in a multimedia communication system for segmenting and multiplexing bitstreams from different media sources into variable length packets, said device comprising:

1A) a plurality of information buffers for receiving bitstreams from different media sources and temporarily buffering these bitstreams;

1B) a dynamic priority-based packet segmentation and multiplexing unit having a multi-discipline queuing scheme, operably coupled to the plurality of information buffers and to an output channel, for selecting and segmenting information bitstreams into variable length packets , wherein said packet size is dynamically adjusted based at least on a fullness of each of the information buffers and a bit-rate of the output channel, and transmitting the packets to an output channel.

2. The multimedia multiplexing device of claim 1 wherein at least one of 2A-2B:

2A) the dynamic priority-based packet segmentation and multiplexing unit having a multi-discipline queuing scheme comprises:

2A1) a buffer monitor, operably coupled to the plurality of information buffers, for monitoring the fullness of each information buffer and sending a packet-transmission request to a queuing and segmentation controller when an information buffer has one packet's worth of information bits ready to be sent;

2A2) the queuing and segmentation controller, operably coupled to the buffer monitor, to the output channel, and to a packet generator, for receiving packet-transmission request(s) from the buffer monitor, for receiving channel bit-rate information from the output channel, for receiving an end of

packet indication from a packet generator, and for modifying a service buffer of a server in accordance with a selected queuing discipline;

2A3) the server, operably coupled to the queuing and segmentation controller and to the plurality of information buffers, for receiving a command from the queuing and segmentation controller and for receiving a bitstream from a selected information buffer and passing the selected bitstream to a packet generator; and

2A4) the packet generator, operably coupled to receive a bitstream from the server, for forming packets and sending packets to the output channel and informing the queuing and segmentation controller each time a packet is sent;

and where selected,

2A5) wherein, in the multi-discipline queuing scheme, where there are N buffers, represented as B_1, B_2,..., and B_N, for storing source bitstream 1, 2,..., and N respectively, a priority of each buffer is the same as the priority of the corresponding bitstream, the buffers are served according to one of 2A5a-2A5c:

2A5a) a Head of Line Priority for Variable Length Packet Segmentation, HOLP-VLP, queuing discipline;

2A5b) a Weighted Round Robin for Variable Length Packet Segmentation, WRR-VLP, queuing discipline; and

2A5c) a predetermined combination of the HOLP-VLP and WRR-VLP queuing disciplines,

and where further selected, at least one of 2A6a-2A6c:

2A6a) the HOLP-VLP queuing discipline is utilized to treat priority groups that all have different priorities and includes the steps of 2A6a1-2A6A2:

2A6a1) where buffer i, i an index of the buffer, is currently being served, continuing, by a server, to serve buffer i until one of the following two events happens:

2A6a1A) at least one packet from a higher priority buffer is ready to be sent;

2A6a1B) an insufficient number of bits are available in buffer i; and

2A6a2) upon one of A1-A2 occuring, stopping serving buffer i, by the server, after completely sending a packet currently being sent and then switching to serve a next highest priority buffer that has packet(s) ready to be sent;

2A6b) the WRR-VLP queuing discipline is utilized to treat bitstreams that all have an equal priority and includes the steps of:

2A6b1) serving, by the server, in a pre-selected partition period, T_p, each buffer in a same priority group cyclically in a predetermined order and for a period, T_i, i an index of the buffer, where the period T_i is one of 2A6b1A-2A6b1B:

2A6b1A) $a_i{}^*T_p$, where a_i is the bandwidth-weighting factor used for bitstream i; and

2A6b1B) shortening T_i where one of 2A61b1B1-2A61b1B2:

2A61b1B1) where insufficient bits are in buffer i; and

2A61b1B2) where a packet from a higher priority buffer is ready to be sent; and

2A6b1B) extending T_i where buffer i still has bits to be sent, but all other buffers in a same priority group are unready to send bits, where an upper bound of a partition period T_p is determined by one of 2A6b1B1-2A6b1B2:

2A6b1B1) a predetermined maximum queuing delay requirement for each bitstream in the priority group and a lower bound of T_p is determined by the packetization efficiency requirement; and

2A6b1B2) T_p is adjusted dynamically;

where, when a particular buffer is allocated a period of $a_i \cdot Tp$ to send a packet, but the packet is interrupted by a packet-transmission request from a higher priority buffer, then a remaining credit is given to the same buffer when the priority group is served again in a following time; and

2A6c) a packet from bitstream i and generated at time t contains a number of raw information bits denoted as $S_i(t)$, excluding packet header, flags and any stuffing bits, where $S_i(t)$ is computed as follows:

if $(Q_i(t) < S_i^{min})$

$S_i(t) = 0$

else

$S_i(t) = \min \{ S_i^{max}, Q_i(t)+Q_i(t_{itp}-t), B_i(t,t_{itp})\}$

where S_i^{min} and S_i^{max}, with $0 < S_i^{min} <= S_i^{max}$, are respectively a minimum and maximum number of raw information bits in a packet for bitstream i, $Q_i(t)$ a number of raw information bits available in buffer i at time t, $Q_i(t_{itp}-t)$ a number of raw information bits entering buffer i between time t and t_{itp}, and $B_i(t,t_{itp})$ a maximum number of bits, that can be sent during a time when said buffer is being continuously served, subtracted by any overhead bits, denoted $O_i(t)$, for the packet, where t_{itp} is an interrupt time at which transmission of the current packet must be stopped, and, where selected, where a fixed number of information bits in each packet is sent, $S_i^{min} = S_i^{max}$, and where selected, one of:

2A6c1) wherein the minimum and maximum number of raw information bits are pre-determined based on efficiency and memory requirements for a given application; and

Page missing upon filing

THIS PAGE BLANK (USPTO)

3. A multimedia multiplexing device in a multimedia communication system for segmenting and multiplexing bitstreams from different media sources into variable length packets, said device comprising:

3A) a plurality of information buffers for receiving bitstreams from the different media sources and temporarily buffering the bitstreams;

3B) a dynamic priority-based packet segmentation and multiplexing unit having a multi-discipline queuing scheme, operably coupled to the plurality of information buffers and to an output channel, for:

3B1) dynamically adjusting sizes of packets for the bitstreams based on a fullness of each of the plurality of information buffers and an available bit-rate of the output channel; and

3B2) multiplexing the packets in an order based on the multi-discipline queuing scheme that assigns a higher priority to delay-sensitive source(s) than to delay-insensitive source(s) and provides an effective bandwidth sharing among a plurality of sources.

4. The multimedia multiplexing device of claim 3 wherein at least one of 4A-4C:

4A) the dynamic priority-based packet segmentation and multiplexing unit comprises :

4A1) a buffer monitor, operably coupled to the plurality of information buffers, for monitoring the fullness of each information buffer and sending a packet-transmission request to a queuing and segmentation controller when an information buffer has one packet worth of information bits ready to be sent;

4A2) the queuing and segmentation controller, operably coupled to the buffer monitor, to the output channel, and to a packet generator, for receiving packet-transmission request(s) from the buffer monitor, for receiving channel bit-rate information from the output channel, for receiving an end of packet indication from a packet generator, and for modifying a service buffer of a server in accordance with a selected queuing discipline;

4A3) the server, operably coupled to the queuing and segmentation controller and to the plurality of information buffers, for receiving a command from the queuing and segmentation controller and for receiving a bitstream from a selected information buffer and passing the selected bitstream to a packet generator; and

4A4) the packet generator, operably coupled to receive a bitstream from the server, for forming packets and sending packets to the output channel and informing the queuing and segmentation controller each time a packet is sent;

4B) the multi-discipline queuing scheme includes prioritizing bitstreams from different sources based on delay tolerances, where a least delay-tolerant bitstream is given the highest priority, a most delay-tolerant bitstream is given the

lowest priority, and bitstreams that have an equal delay tolerance are given a same priority and are grouped into a single priority group, and where selected, wherein a bandwidth-weighting factor a_i for bitstream i, i an index of the bitstream, is assigned for each bitstream as a fraction of bandwidth allocated to the bitstream i out of the total bandwidth allocated to the priority group containing bitstream i;

and

4C) wherein, in the multi-discipline queuing scheme, where there are N buffers, represented as B_1, B_2,..., and B_N, for storing source bitstream 1, 2,..., and N respectively, a priority of each buffer is the same as the priority of the corresponding bitstream, the buffers are served according to one of 4C1-4C3:

4C1) a Head of Line Priority for Variable Length Packet Segmentation, HOLP-VLP, queuing discipline;

4C2) a Weighted Round Robin for Variable Length Packet Segmentation, WRR-VLP, queuing discipline; and

4C3) a predetermined combination of the HOLP-VLP and WRR-VLP queuing disciplines,

and where selected at least one of 4C3a-4C3c:

4C3a) a HOLP-VLP queuing discipline is utilized to treat priority groups that all have different priorities and includes the steps of:

4C3a1) where buffer i, i an index of the buffer, is currently being served, continuing, by a server, to serve buffer i until one of the following two events happens:

4C3a1A) at least one packet from a higher priority buffer is ready to be sent;

4C3a1B) an insufficient number of bits are available in buffer i;

4C3a2) upon one of A1-A2 occuring, stopping serving buffer i, by the server, after completely sending

a packet currently being sent and then switching to serve a next highest priority buffer that has packet(s) ready to be sent;

4C3b) a WRR-VLP queuing discipline is utilized to treat bitstreams that all have an equal priority and includes the steps:

4C3b1) serving, by the server, in a pre-selected partition period, T_p, each buffer in a same priority group cyclically in a predetermined order and for a period, T_i, i an index of the buffer, where the period T_i is one of 4C3b1A-4C3b1C:

4C3b1A1) $a_i{}^*T_p$, where a_i is the bandwidth weighting factor used for bitstream i; and

4C3b1A2) shortening T_i where one of A2a-A2b:

4C3b1A2a) where insufficient bits is in buffer i; and

4C3b1A2a) where a packet from a higher priority buffer is ready to be sent; and

4C3b1C) extending T_i where buffer i still has bits to be sent, but all other buffers in a same priority group are unready to send bits, where an upper bound of a partition period T_p is determined by one of 4C3b1C1-4C3b1C2:

4C3b1C1) a predetermined maximum queuing delay requirement for each bitstream in the priority group and a lower bound of T_p is determined by the packetization efficiency requirement; and

4C3b1C2) T_p is adjusted dynamically;

where, when a particular buffer is allocated a period of $a_i{}^*T_p$ to send a packet, but the packet is interrupted by a packet-transmission request from a higher priority buffer, then a remaining credit is given to a same buffer when the priority group is served again in a following time; and

4C3c) a packet from bitstream i and generated at time t contains the number of raw information bits denoted as $S_i(t)$, excluding packet header, flags and any stuffing bits, where $S_i(t)$ is computed as follows:

if $(Q_i(t) < S_i^{min})$

$S_i(t) = 0$

else

$S_i(t) = \min \{ S_i^{max}, Q_i(t)+Q_i(t_{itp}-t), B_i(t,t_{itp}) \}$

where S_i^{min} and S_i^{max}, with $0 < S_i^{min} <= S_i^{max}$, are respectively a minimum and maximum number of raw information bits in a packet for bitstream i, $Q_i(t)$ a number of raw information bits available in buffer i at time t, $Q_i(t_{itp}-t)$ a number of raw information bits entering buffer i between time t and t_{itp}, and $B_i(t,t_{itp})$ a maximum number of bits, that can be sent during a time when said buffer is being continuously served, subtracted by any overhead bits, denoted $O_i(t)$, for the packet, where t_{itp} is the interrupt time at which transmission of the current packet must be stopped, and, where selected, where a fixed number of information bits in each packet is sent, $S_i^{min} = S_i^{max}$, and where selected one of 4C3c1-4C3c2:

4C3c1) wherein the minimum and maximum number of raw information bits are pre-determined based on efficiency and memory requirements for a given application; and

4C3c2) wherein flags and stuffing bits for a packet are generated by a hardware HDLC controller, and the available bandwidth $B_i(t,t_{itp})$ is estimated as a channel bit-rate multiplied by a buffer serving time, $t-t_{itp}$, and then is subtracted by overhead corrections due to an underestimate of overhead of packets sent previously, where a correction for a packet is computed as T*R-S, where:

4C3c2A) T represents a time difference between an acknowledgment to the packet and that of an immediately preceeding packet;

4C3c2B) R represents the channel bit-rate; and

4C3c2)C) S represents a number of known information bits in the packet.

5. A multimedia multiplexing method for a multimedia communication system for segmenting and multiplexing bitstreams from different media sources into variable length packets, said method comprising the steps of:

5A) receiving bitstreams from different media sources and temporarily buffering the bitstreams in a plurality of information buffers;

5B) utilizing a dynamic priority-based packet segmentation and multiplexing unit having a multi-discipline queuing scheme for:

5B1) dynamically adjusting sizes of packets for the information bitstreams based on a fullness of each of the plurality of information buffers and an available bit-rate of the output channel and

5B2) multiplexing the packets in an order based on the multi-discipline queuing scheme that assigns a higher priority to delay-sensitive source(s) than to delay-insensitive source(s) and provides an effective bandwidth sharing among a plurality of sources.

6. The method of claim 5 wherein at least one of 6A-6C:

6A) the dynamic priority-based packet segmentation and multiplexing unit having a multi-discipline queuing scheme utilizes the steps of:

6A1) monitoring, by a buffer monitor, the fullness of each information buffer and sending a packet-transmission request to a queuing and segmentation controller when an information buffer has one packet worth of information bits ready to be sent;

6A2) utilizing a queuing and segmentation controller for receiving packet transmission request(s) from the buffer monitor, for receiving channel bit-rate information from an output channel, for receiving an end of packet indication from a packet generator, and for modifying a service buffer of a server in accordance with a selected queuing discipline;

6A3) utilizing a server for receiving a command from the queuing and segmentation controller and for receiving a bitstream from a selected information buffer and passing the selected bitstream to a packet generator; and

6A4) utilizing a packet generator for forming packets and sending the packets to the output channel and informing the queuing and segmentation controller each time a packet is sent;

6B) the multi-discipline queuing scheme includes prioritizing bitstreams from different sources based on delay tolerances, where a least delay-tolerant bitstream is given the highest priority, a most delay-tolerant bitstream is given the lowest priority, and bitstreams that have an equal delay tolerance are given a same priority and are grouped into a single priority group, and where selected, wherein a bandwidth-weighting factor a_i for a bitstream i, i an index of the bitstream, is assigned for each bitstream as a fraction of bandwidth allocated to the bitstream i out of the total bandwidth allocated to the priority group containing bitstream i; and

6C) wherein, in the multi-discipline queuing scheme, where there are N buffers, represented as B_1, B_2,..., and B_N, for storing source bitstream 1, 2,..., and N respectively, a priority of each

buffer is the same as the priority of the corresponding bitstream, the buffers are served according to one of 6C1-6C3:

6C1) a Head of Line Priority for Variable Length Packet Segmentation, HOLP-VLP, queuing discipline;

6C2) a Weighted Round Robin for Variable Length Packet Segmentation, WRR-VLP, queuing discipline; and

6C3) a predeterimined combination of the HOLP-VLP and WRR-VLP queuing disciplines;

and where selected, at least one of 6C3a-6C3d:

6C3a) wherein the HOLP-VLP queuing discipline is utilized to treat priority groups that all have different priorities and includes the steps:

6C3a1) where buffer i, i an index of the buffer, is currently being served, continuing, by a server, to serve buffer i until one of the following two events happens:

6C3a1A) at least one packet from a higher priority buffer is ready to be sent;

6C3a1B) an insufficient number of bits are available to in buffer i;

6C3a2) upon one of the events occuring, stopping serving buffer i, by the server, after completely sending a packet currently being sent and then switching to serve a next highest priority buffer that has packet(s) ready to be sent;

6C3b) wherein, the WRR-VLP queuing discipline is utilized to treat bitstreams that all have an equal priority and includes the steps:

6C3b1) serving, by the server, in a pre-selected partition period, T_p, each buffer in a same priority group cyclically in a predetermined order and for a period, T_i, i an index of the buffer, where the period T_i is one of 6C3b1A-6C3b1C:

6C3b1A) $a_i{}^*T_p$, where a_i is the bandwidth weighting factor used for bitstream i; and

6C3b1B) shortening T_i where one of 6C3b1B1-6C3b1B2:

6C3b1B1) where insufficient bits are in buffer i; and

6C3b1B1) where a packet from a higher priority buffer is ready to be sent; and

6C3b1C) extending T_i where buffer i still has bits to be sent, but all other buffers in a same priority group are unready to send bits, where an upper bound of a partition period T_p is determined by one of 6C3b1C1-6C3b1C2:

6C3b1C1) a predetermined maximum queuing delay requirement for each bitstream in the priority group and a lower bound of T_p is determined by the packetization efficiency requirement; and

6C3b1C2) T_p is adjusted dynamically;

where, when a particular buffer is allocated a period of $a_i{}^*Tp$ to send a packet, but the packet is interrupted by a packet-transmission request from a higher priority buffer, then the remaining credit is given to the same buffer when the priority group is served again in a following time;

6C3c) wherein a packet from bitstream i and generated at time t contains a number of raw information bits denoted as $S_i(t)$, excluding packet header, flags and any stuffing bits, where $S_i(t)$ is computed as follows:

if $(Q_i(t) < S_i{}^{min})$

$S_i(t) = 0$

else

$S_i(t) = \min \{ S_i{}^{max}, Q_i(t)+Q_i(t_itp\text{-}t), B_i(t,t_itp)\}$

where S_i^{min} and S_i^{max}, with $0 < S_i^{min} <= S_i^{max}$, are respectively a minimum and maximum number of raw information bits in a packet for bitstream i, $Q_i(t)$ number of raw information bits available in buffer i at time t, $Q_i(t_{itp}\text{-}t)$ a number of raw information bits entering buffer i between time t and t_{itp}, and $B_i(t,t_{itp})$ a maximum number of bits, that can be sent during a time when said buffer is being continuously served, subtracted by any overhead bits, denoted $O_i(t)$, for the packet, where t_{itp} is the interrupt time at which transmission of the current packet must be stopped, and, where selected, where a fixed number of information bits in each packet is sent, $S_i^{min} = S_i^{max}$, and where selected, at least one of 6C3c1-6C3c2:

6C3c1) wherein the minimum and maximum number of raw information bits are pre-determined based on efficiency and memory requirements for a given application; and where further selected, wherein flags and stuffing bits for a packet are generated by a hardware HDLC controller, and the available bandwidth $B_i(t,t_{itp})$ is estimated as a channel bit-rate multiplied by a buffer serving time, $t\text{-}t_{itp}$, and then is subtracted by the overhead corrections due to an underestimate of overhead of packets sent previously, where a correction for a packet is computed as T*R-S, where:

6C3c1A) T represents a time difference between an acknowledgment to the packet and that of an immediately preceeding packet;

6C3cB) R represents the channel bit-rate; and

6C3cC) S represents a number of known information bits in the packet; and

6C3d) the multi-discipline queuing scheme includes, where a real-time audio bitstream, denoted as bitstream 1, a real-time video bitstream, denoted as bitstream 2, a real-time data

bitstream, denoted as bitstream 3, and a non-real-time data bitstream, denoted as a bitstream 4, are multiplexed together, making priority assignments with P1 > P2 = P3 > P4, where Pi is a priority associated with bitstream i, and i = 1, 2, 3, 4, assigning bitstream 2 and 3 in a same priority group, denoted as PG2, that are multiplexed according to the WRR-VLP queuing discipline, and then multiplexing bitstream 1, PG2, and bitstream 4 according to the HOLP-VLP queuing discipline.

7. Data Communications Equipment/Data Terminal Equipment having a multimedia multiplexing device in a multimedia communication system for segmenting and multiplexing bitstreams from different media sources into variable length packets, said multimedia multiplexing device comprising:

7A) a plurality of information buffers for receiving bitstreams from the different media sources and temporarily buffering the bitstreams;

7B) a dynamic priority-based packet segmentation and multiplexing unit having a multi-discipline queuing scheme, operably coupled to the plurality of information buffers and to an output channel, for:

7B1) dynamically adjusting sizes of packets for the bitstreams based on a fullness of each of the plurality of information buffers and an available bit-rate of the output channel; and

7B2) multiplexing the packets in an order based on the multi-discipline queuing scheme that assigns a higher priority to delay-sensitive source(s) than to delay-insensitive source(s) and provides an effective bandwidth sharing among a plurality of sources.

8. The Data Communications Equipment/Data Terminal Equipment of claim 7 wherein the dynamic priority-based packet segmentation and multiplexing unit comprises:

8A) a buffer monitor, operably coupled to the plurality of information buffers, for monitoring the fullness of each information buffer and sending a packet-transmission request to a queuing and segmentation controller when an information buffer has one packet worth of information bits ready to be sent;

8B) the queuing and segmentation controller, operably coupled to the buffer monitor, to the output channel, and to a

packet generator, for receiving packet-transmission request(s) from the buffer monitor, for receiving channel bit-rate information from the output channel, for receiving an end of packet indication from a packet generator, and for modifying a service buffer of a server in accordance with a selected queuing discipline;

8C) the server, operably coupled to the queuing and segmentation controller and to the plurality of information buffers, for receiving a command from the queuing and segmentation controller and for receiving a bitstream from a selected information buffer and passing the selected bitstream to a packet generator; and

8D) the packet generator, operably coupled to receive a bitstream from the server, for forming packets and sending packets to the output channel and informing the queuing and segmentation controller each time a packet is sent.

9. The Data Communications Equipment/Data Terminal Equipment of claim 7 wherein the multi-discipline queuing scheme includes prioritizing bitstreams from different sources based on delay tolerances, where a least delay-tolerant bitstream is given the highest priority, a most delay-tolerant bitstream is given the lowest priority, and bitstreams that have an equal delay tolerance are given a same priority and are grouped into a single priority group, and where selected, wherein a bandwidth-weighting factor a_i for bitstream i, i an index of the bitstream, is assigned for each bitstream as a fraction of bandwidth allocated to the bitstream i out of the total bandwidth allocated to the priority group containing bitstream i.

10. The Data Communications Equipment/Data Terminal Equipment of claim 7 wherein, in the multi-discipline queuing

scheme, where there are N buffers, represented as B_1, B_2,..., and B_N, for storing source bitstream 1, 2,..., and N respectively, a priority of each buffer is the same as the priority of the corresponding bitstream, the buffers are served according to one of 10A-10C:

10A) a Head of Line Priority for Variable Length Packet Segmentation, HOLP-VLP, queuing discipline;

10B) a Weighted Round Robin for Variable Length Packet Segmentation, WRR-VLP, queuing discipline; and

10C) a predetermined combination of the HOLP-VLP and WRR-VLP queuing disciplines,

and where selected, at least one of 10C1-10C3:

10C1) wherein, the HOLP-VLP queuing discipline is utilized to treat priority groups that all have different priorities and includes the steps of:

10C1A) where buffer i, i an index of the buffer, is currently being served, continuing, by a server, to serve buffer i until one of the following two events happens:

10C1A1) at least one packet from a higher priority buffer is ready to be sent;

10C1A2) an insufficient number of bits are available in buffer i;

10C1B) upon one of 10C1A1-10C1A2 occuring, stopping serving buffer i, by the server, after completely sending a packet currently being sent and then switching to serve a next highest priority buffer that has packet(s) ready to be sent;

10C2) wherein, the WRR-VLP queuing discipline is utilized to treat bitstreams that all have an equal priority and includes the steps:

10C2A) serving, by the server, in a pre-selected partition period, T_p, each buffer in a same priority group

cyclically in a predetermined order and for a period, T_i, i an index of the buffer, where the period T_i is one of 10C2A1-10C2A3:

10C2A1) a_i*T_p, where a_i is the bandwidth weighting factor used for bitstream i; and

10C2A2) shortening T_i where one of 10C2A2a-10C2A2b:

10C2A2a) where insufficient bits is in buffer i; and

10C2A2b) where a packet from a higher priority buffer is ready to be sent; and

10C2A3) extending T_i where buffer i still has bits to be sent, but all other buffers in a same priority group are unready to send bits, where an upper bound of a partition period T_p is determined by one of 10C2A3a-10C2A3b:

10C2A3a) a predetermined maximum queuing delay requirement for each bitstream in the priority group and a lower bound of T_p is determined by the packetization efficiency requirement; and

10C2A3b) T_p is adjusted dynamically;

where, when a particular buffer is allocated a period of a_i*Tp to send a packet, but the packet is interrupted by a packet-transmission request from a higher priority buffer, then a remaining credit is given to a same buffer when the priority group is served again in a following time; and

10C3) wherein a packet from bitstream i and generated at time t contains the number of raw information bits denoted as $S_i(t)$, excluding packet header, flags and any stuffing bits, where $S_i(t)$ is computed as follows:

if ($Q_i(t) < S_i^{min}$)

$S_i(t) = 0$

else

$$S_i(t) = \min \{ S_i^{max}, Q_i(t)+Q_i(t_{itp}-t), B_i(t,t_{itp})\}$$

where S_i^{min} and S_i^{max}, with $0 < S_i^{min} <= S_i^{max}$, are respectively a minimum and maximum number of raw information bits in a packet for bitstream i, $Q_i(t)$ a number of raw information bits available in buffer i at time t, $Q_i(t_{itp}-t)$ a number of raw information bits entering buffer i between time t and t_{itp}, and $B_i(t,t_{itp})$ a maximum number of bits, that can be sent during a time when said buffer is being continuously served, subtracted by any overhead bits, denoted $O_i(t)$, for the packet, where t_{itp} is the interrupt time at which transmission of the current packet must be stopped, and, where selected, where a fixed number of information bits in each packet is sent, $S_i^{min} = S_i^{max}$, and where selected, at least one of 10C3a-10C3b:

10C3a) wherein the minimum and maximum number of raw information bits are pre-determined based on efficiency and memory requirements for a given application; and

10C3b) wherein flags and stuffing bits for a packet are generated by a hardware HDLC controller, and the available bandwidth $B_i(t,t_{itp})$ is estimated as a channel bit-rate multiplied by a buffer serving time, $t-t_{itp}$, and then is subtracted by the overhead corrections due to an underestimate of overhead of packets sent previously, where a correction for a packet is computed as T*R-S, where:

10C3b1) T represents a time difference between an acknowledgment to the packet and that of an immediately preceeding packet;

10C3b2) R represents the channel bit-rate; and

10C3b3) S represents a number of known information bits in the packet.

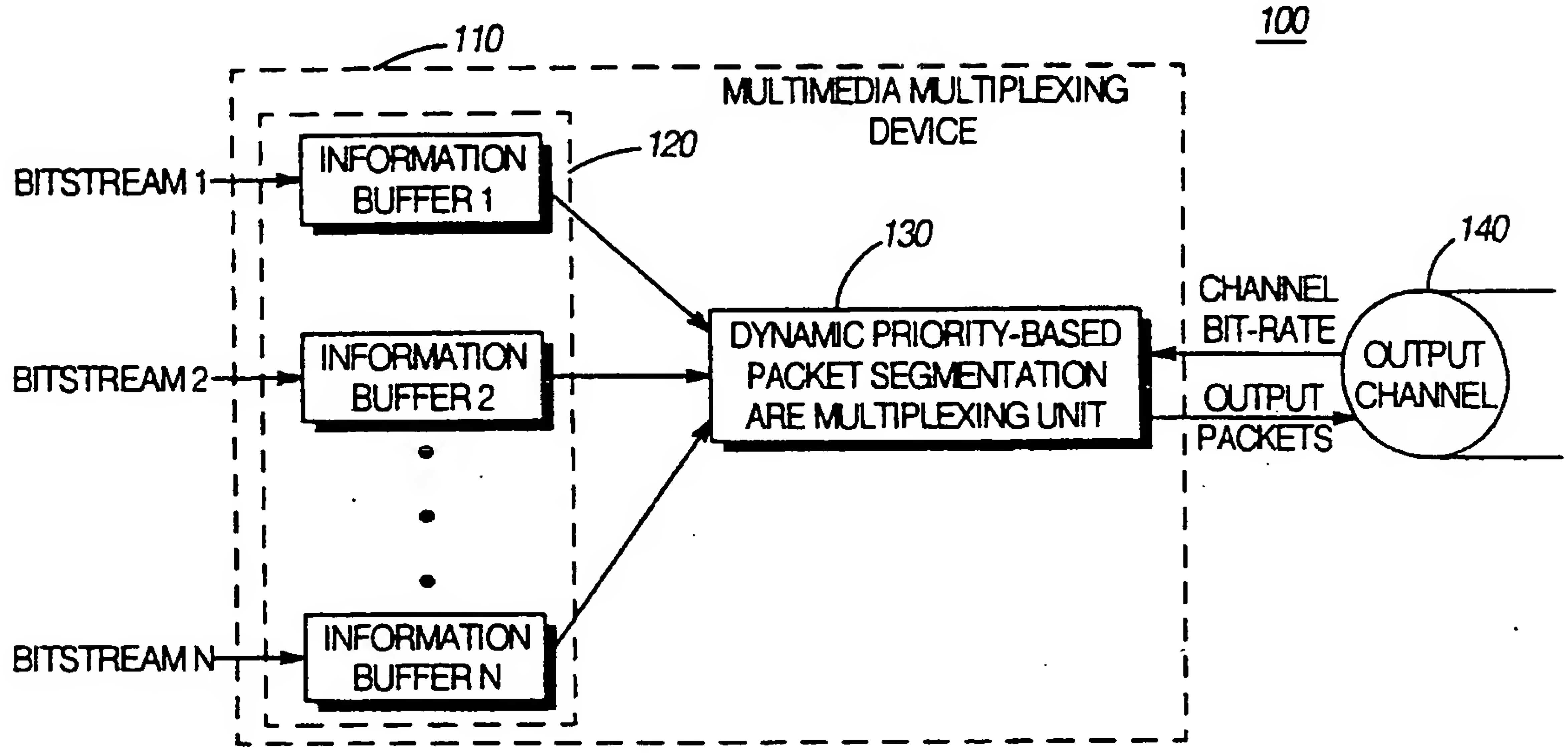

100
110
MULTIMEDIA MULTIPLEXING DEVICE
120
BITSTREAM 1
INFORMATION BUFFER 1
BITSTREAM 2
INFORMATION BUFFER 2
BITSTREAM N
INFORMATION BUFFER N
130
DYNAMIC PRIORITY-BASED PACKET SEGMENTATION ARE MULTIPLEXING UNIT
CHANNEL BIT-RATE
OUTPUT PACKETS
140
OUTPUT CHANNEL

Fig.1

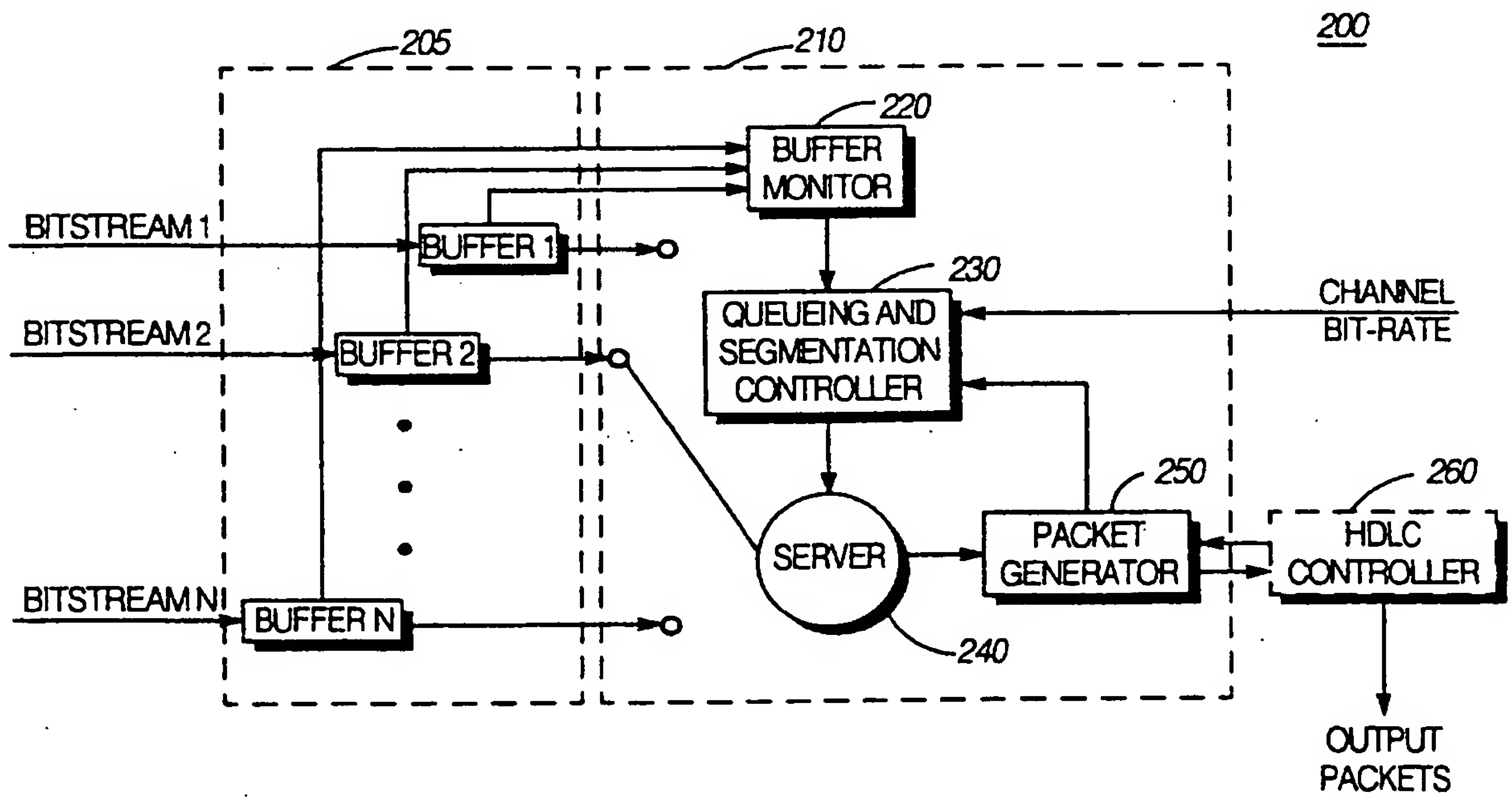

200
205
210
220
BUFFER MONITOR
BITSTREAM 1
BUFFER 1
BITSTREAM 2
BUFFER 2
BITSTREAM N
BUFFER N
230
QUEUEING AND SEGMENTATION CONTROLLER
CHANNEL BIT-RATE
250
260
SERVER
PACKET GENERATOR
HDLC CONTROLLER
240
OUTPUT PACKETS

Fig.2

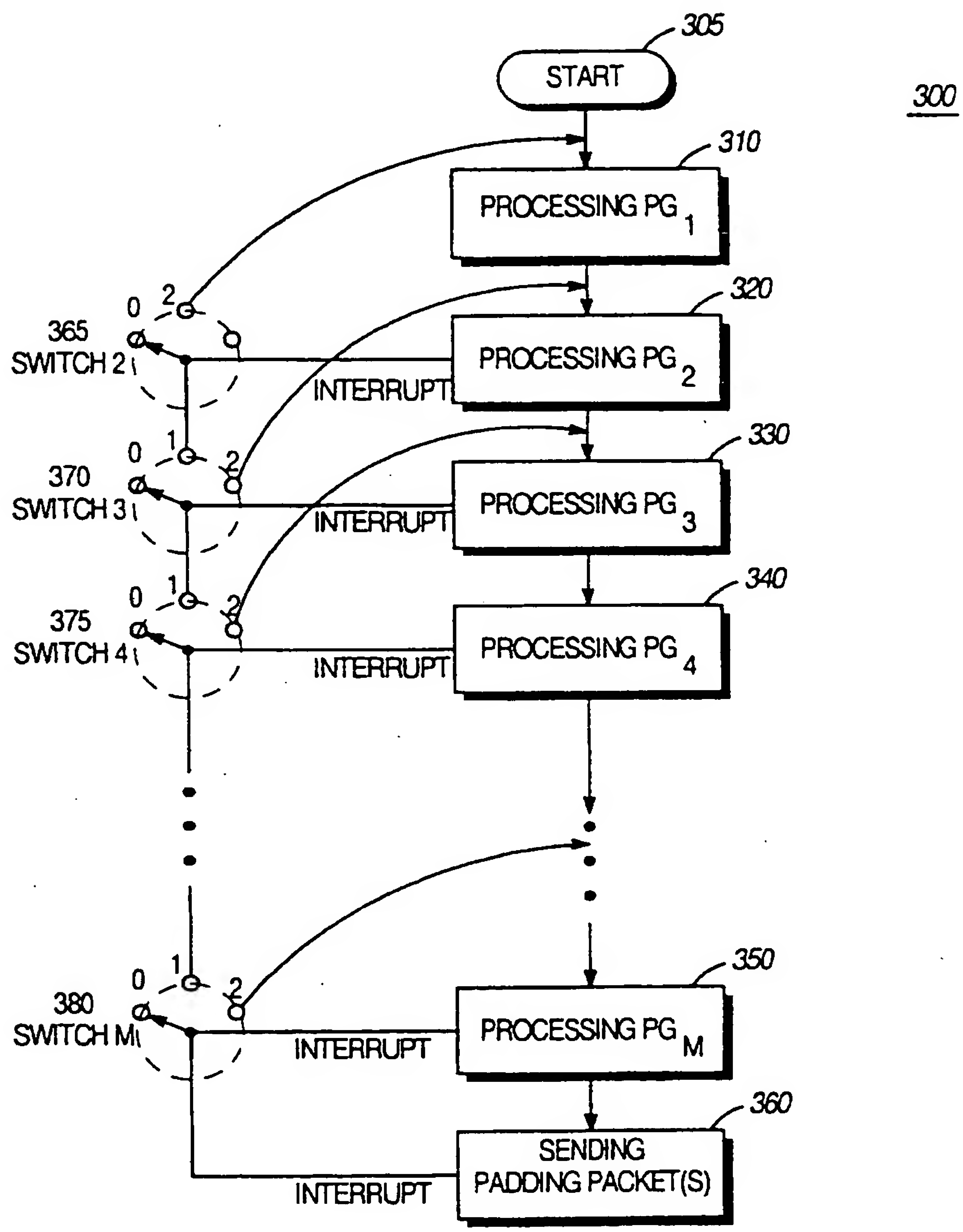

305
START
300
310
PROCESSING PG 1
320
PROCESSING PG 2
330
PROCESSING PG 3
340
PROCESSING PG 4
350
PROCESSING PG M
360
SENDING PADDING PACKET(S)
365 SWITCH 2
370 SWITCH 3
375 SWITCH 4
380 SWITCH M
0
1
2
INTERRUPT

Fig.3

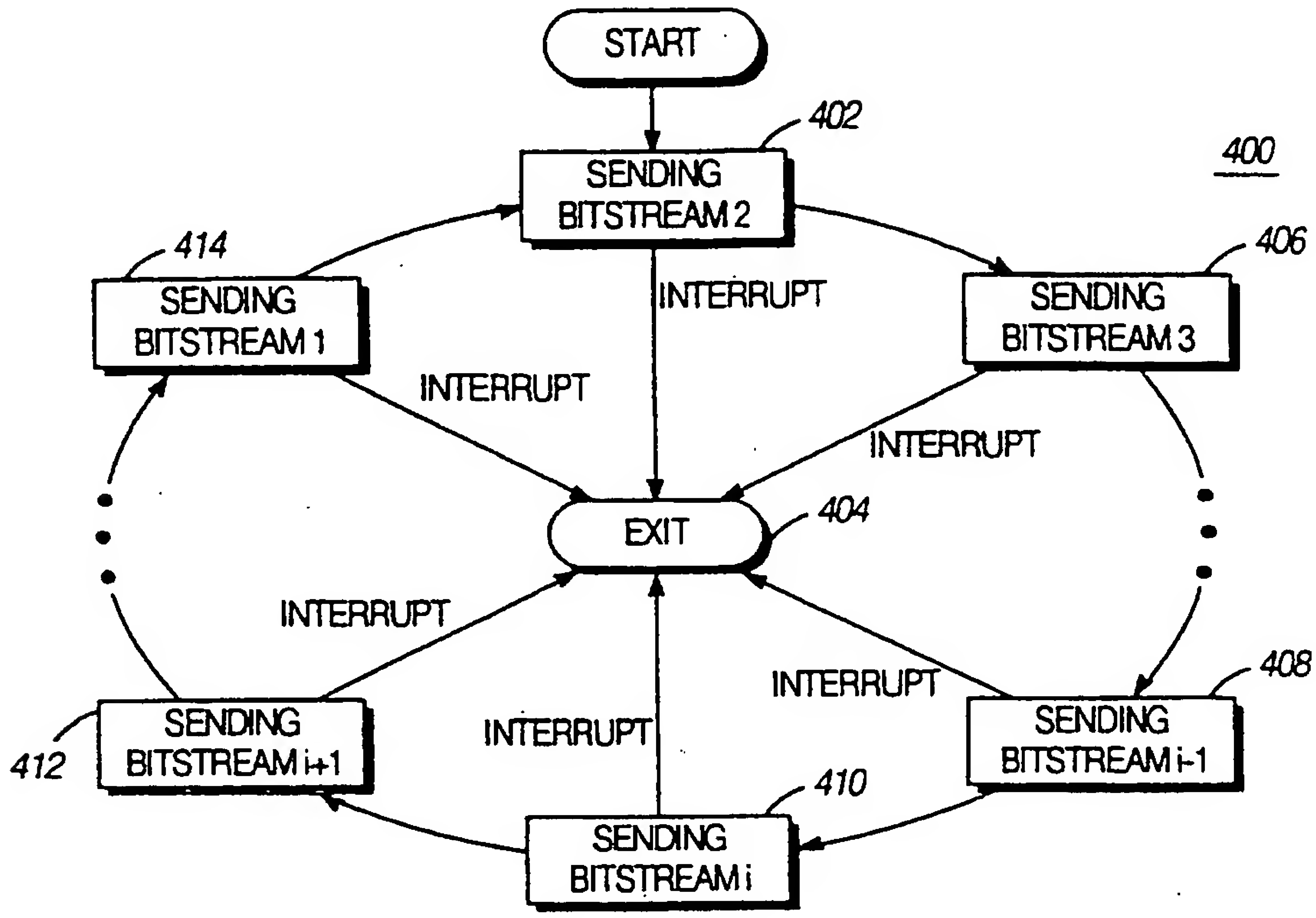


Fig.4

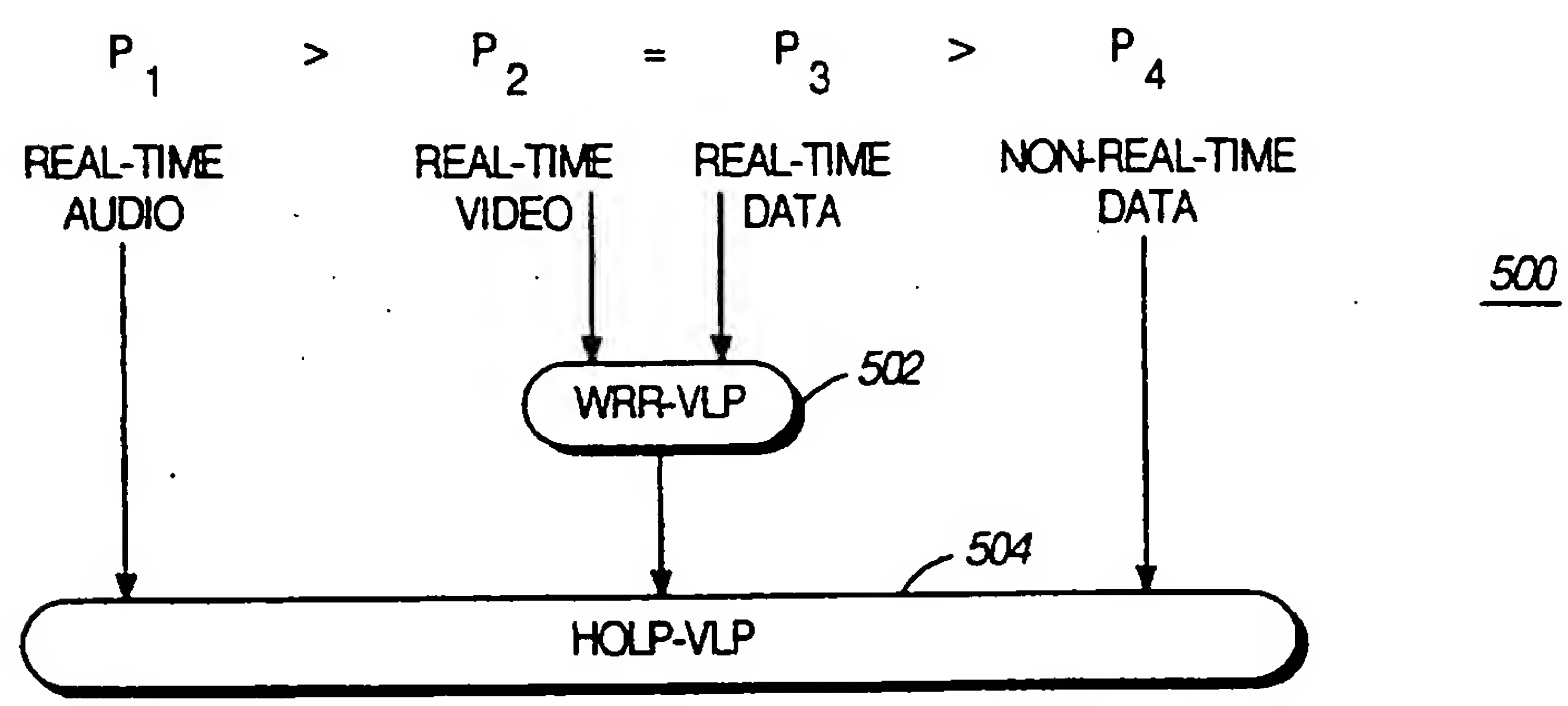


Fig.5

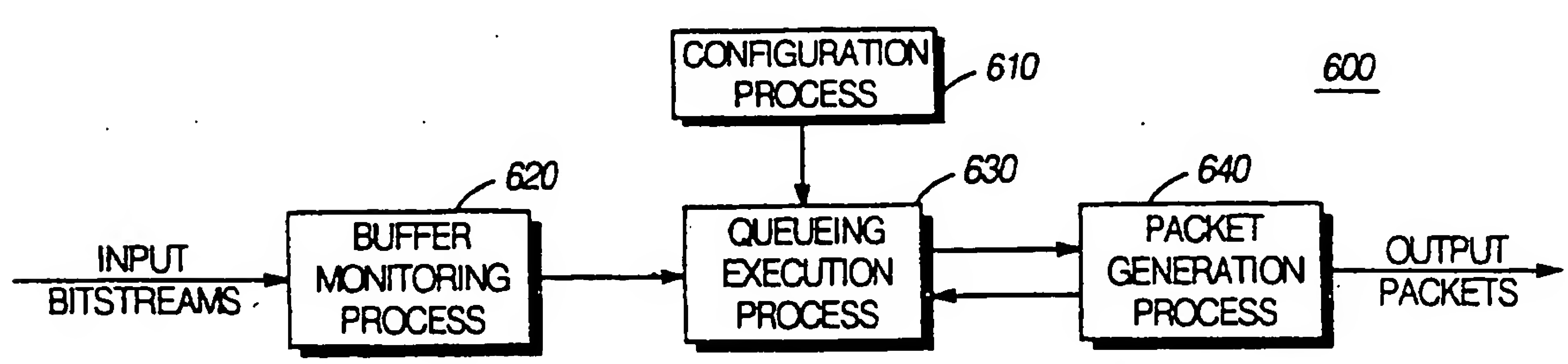


Fig.6

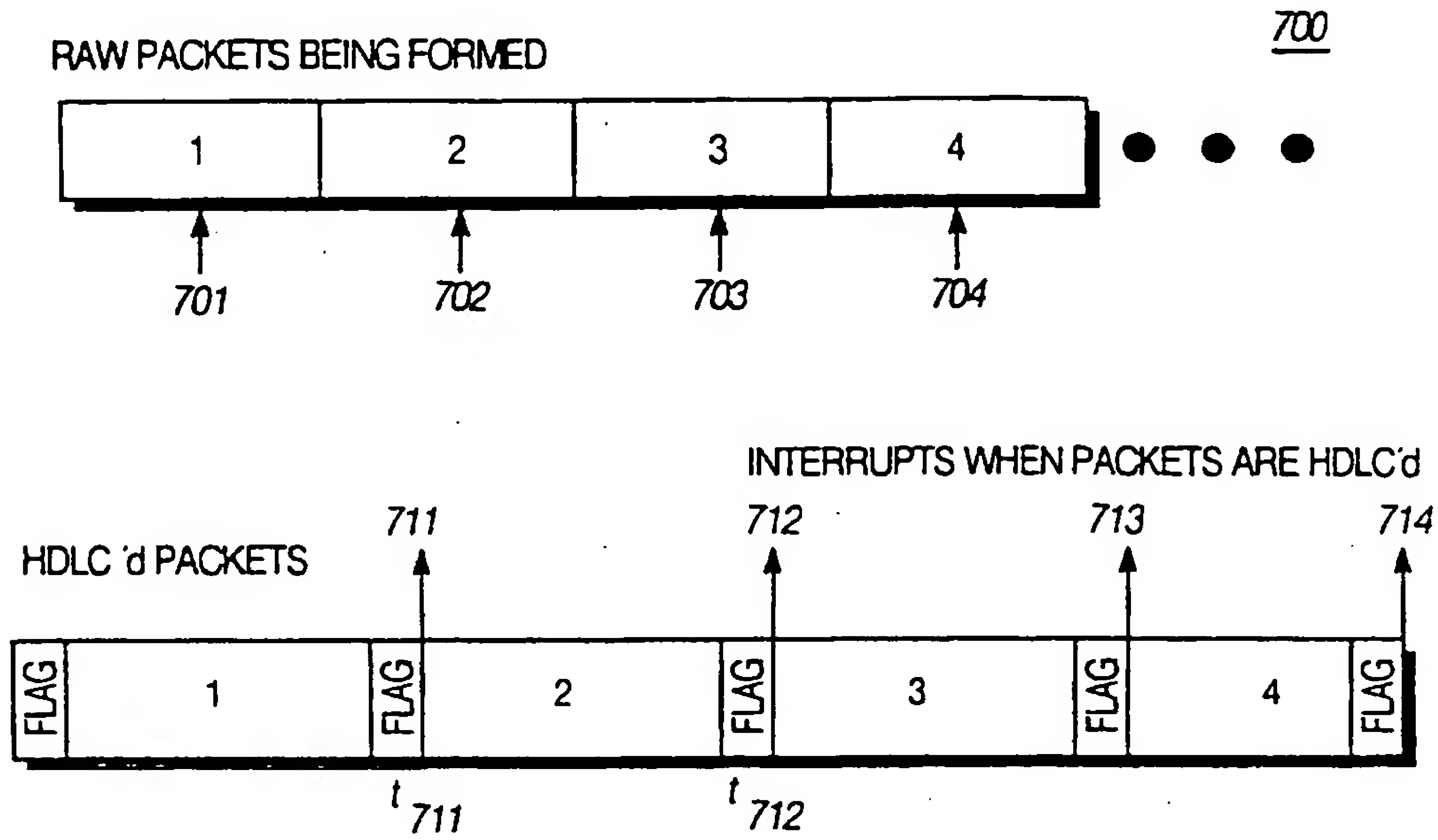

700
RAW PACKETS BEING FORMED
1
2
3
4
701
702
703
704
INTERRUPTS WHEN PACKETS ARE HDLC'd
HDLC 'd PACKETS
711
712
713
714
FLAG
1
FLAG
2
FLAG
3
FLAG
4
FLAG
t 711
t 712

Fig.7

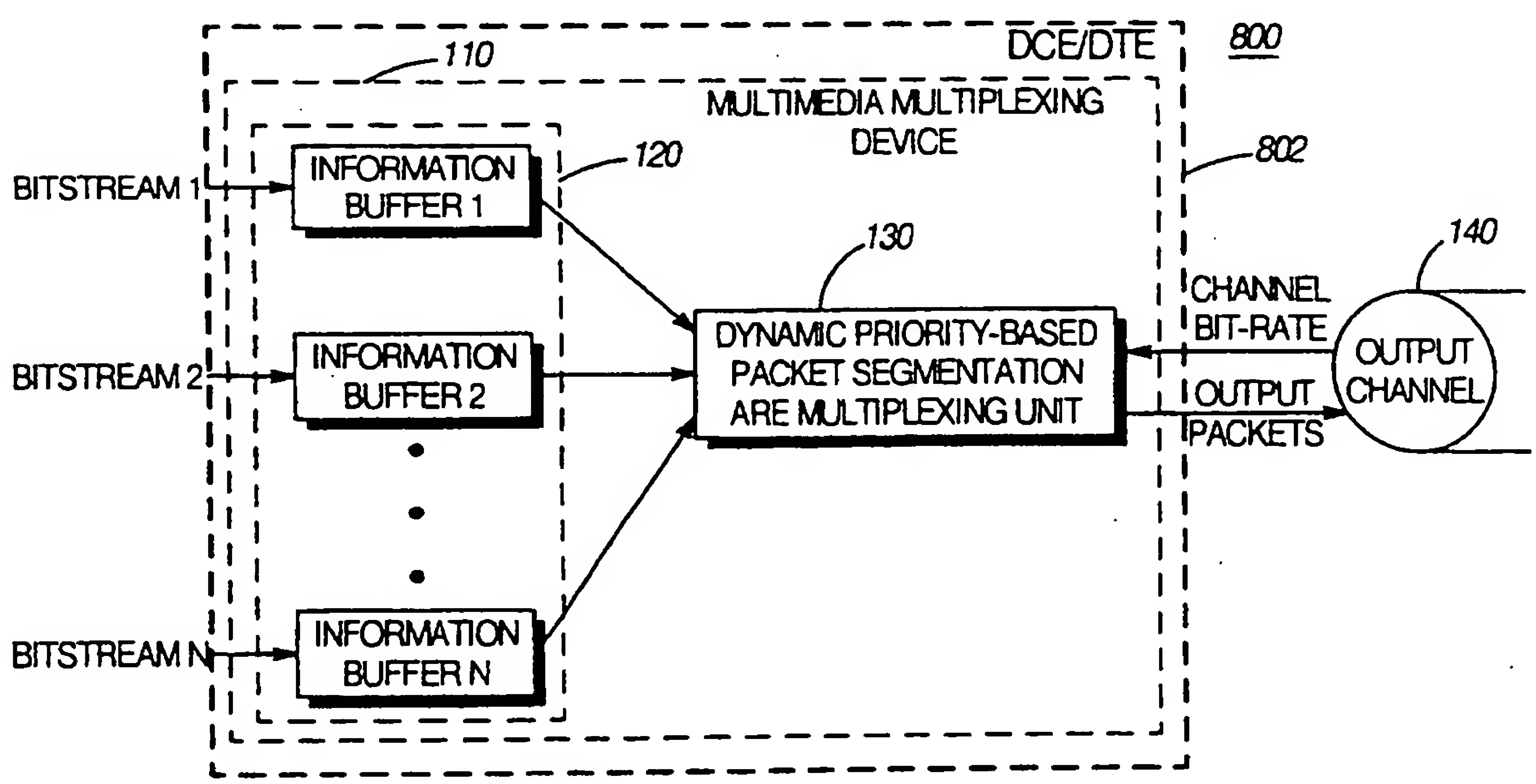

DCE/DTE
800
110
MULTIMEDIA MULTIPLEXING DEVICE
802
120
BITSTREAM 1
INFORMATION BUFFER 1
BITSTREAM 2
INFORMATION BUFFER 2
BITSTREAM N
INFORMATION BUFFER N
130
DYNAMIC PRIORITY-BASED PACKET SEGMENTATION ARE MULTIPLEXING UNIT
CHANNEL BIT-RATE
OUTPUT PACKETS
140
OUTPUT CHANNEL

Fig.8

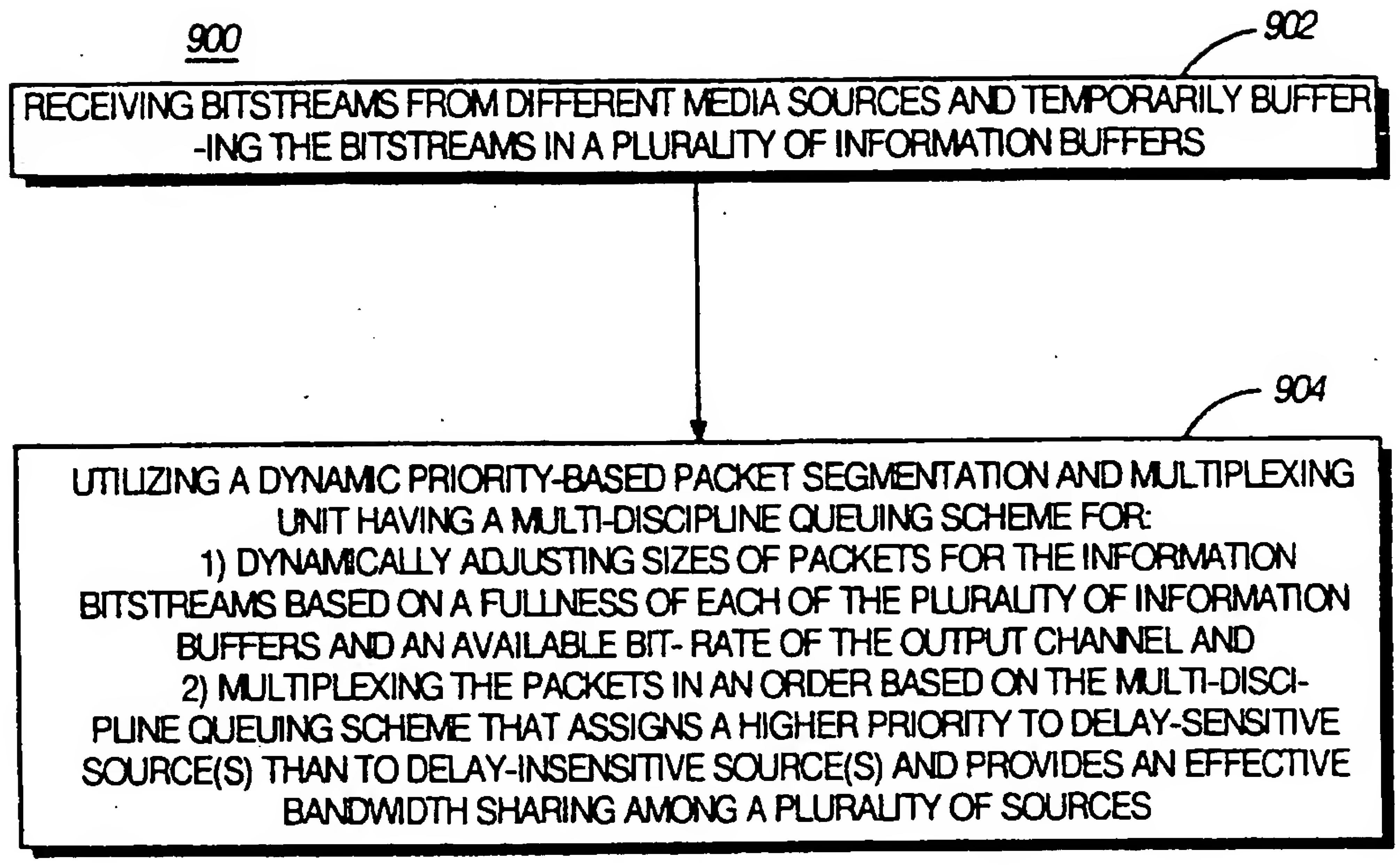

900
902
RECEIVING BITSTREAMS FROM DIFFERENT MEDIA SOURCES AND TEMPORARILY BUFFER -ING THE BITSTREAMS IN A PLURALITY OF INFORMATION BUFFERS
904
UTILIZING A DYNAMIC PRIORITY-BASED PACKET SEGMENTATION AND MULTIPLEXING UNIT HAVING A MULTI-DISCIPLINE QUEUING SCHEME FOR:
1) DYNAMICALLY ADJUSTING SIZES OF PACKETS FOR THE INFORMATION BITSTREAMS BASED ON A FULLNESS OF EACH OF THE PLURALITY OF INFORMATION BUFFERS AND AN AVAILABLE BIT- RATE OF THE OUTPUT CHANNEL AND
2) MULTIPLEXING THE PACKETS IN AN ORDER BASED ON THE MULTI-DISCI- PLINE QUEUING SCHEME THAT ASSIGNS A HIGHER PRIORITY TO DELAY-SENSITIVE SOURCE(S) THAN TO DELAY-INSENSITIVE SOURCE(S) AND PROVIDES AN EFFECTIVE BANDWIDTH SHARING AMONG A PLURALITY OF SOURCES

Fig.9

INTERNATIONAL SEARCH REPORT

International application No.
PCT/US95/14637

A. CLASSIFICATION OF SUBJECT MATTER

IPC(6) :H04J 3/16, 3/26; H04Q 11/04
US CL :U.S. CL.: 370/61, 82, 84, 85.6, 94.1, 112, 118; 340/825.5
According to International Patent Classification (IPC) or to both national classification and IPC

B. FIELDS SEARCHED

Minimum documentation searched (classification system followed by classification symbols)

U.S. : Please See Extra Sheet.

Documentation searched other than minimum documentation to the extent that such documents are included in the fields searched

None

Electronic data base consulted during the international search (name of data base and, where practicable, search terms used)

APS, search terms: (packet# or cell#), (buffer? or queu?), priority, variable length, (segment? or disassembl?), (multiplex? or assembl? or reassembl?), source#, bandwidth(p)allocat?

C. DOCUMENTS CONSIDERED TO BE RELEVANT

Category*	Citation of document, with indication, where appropriate, of the relevant passages	Relevant to claim No.
Y	US, A, 5,132,966 (Hayano et al.) 21 July 1992, col. 2, line 44 to col. 3, line 40; col. 5, line 7 to col. 6, line 55.	1, 3, 5, 7
Y	US, A, 5,251,209 (Jurkvich et al.) 05 October 1993, col. 3, line 2 to col. 7, line 32.	1, 3, 5, 7
A	US, A, 5,140,584 (Suzuki) 18 August 1992, see entire document.	1-8
A	US, A, 5,268,900 (Hluchyj et al.) 07 December 1993, see entire document.	1-8

☐ Further documents are listed in the continuation of Box C. ☐ See patent family annex.

* Special categories of cited documents:

"A" document defining the general state of the art which is not considered to be part of particular relevance

"E" earlier document published on or after the international filing date

"L" document which may throw doubts on priority claim(s) or which is cited to establish the publication date of another citation or other special reason (as specified)

"O" document referring to an oral disclosure, use, exhibition or other means

"P" document published prior to the international filing date but later than the priority date claimed

"T" later document published after the international filing date or priority date and not in conflict with the application but cited to understand the principle or theory underlying the invention

"X" document of particular relevance; the claimed invention cannot be considered novel or cannot be considered to involve an inventive step when the document is taken alone

"Y" document of particular relevance; the claimed invention cannot be considered to involve an inventive step when the document is combined with one or more other such documents, such combination being obvious to a person skilled in the art

"&" document member of the same patent family

Date of the actual completion of the international search	Date of mailing of the international search report
05 JANUARY 1996	15 FEB 1996

Name and mailing address of the ISA/US
Commissioner of Patents and Trademarks
Box PCT
Washington, D.C. 20231
Facsimile No. (703) 305-3230

Authorized officer
ALPUS H. HSU
Telephone No. (703) 305-4377

B. FIELDS SEARCHED
Minimum documentation searched
Classification System: U.S.

U.S. CL.: 370/56, 58.1-58.3, 60, 61, 79, 82, 84, 85.6., 85.7, 94.1, 95.1, 110.1, 112, 118; 340/825.5, 825.51, 825.52; 395/153, 154, 250, 500, 800

This Page is Inserted by IFW Indexing and Scanning Operations and is not part of the Official Record

BEST AVAILABLE IMAGES

Defective images within this document are accurate representations of the original documents submitted by the applicant.

Defects in the images include but are not limited to the items checked:

- ☐ BLACK BORDERS
- ☐ IMAGE CUT OFF AT TOP, BOTTOM OR SIDES
- ☐ FADED TEXT OR DRAWING
- ☐ BLURRED OR ILLEGIBLE TEXT OR DRAWING
- ☐ SKEWED/SLANTED IMAGES
- ☐ COLOR OR BLACK AND WHITE PHOTOGRAPHS
- ☐ GRAY SCALE DOCUMENTS
- ☑ LINES OR MARKS ON ORIGINAL DOCUMENT
- ☐ REFERENCE(S) OR EXHIBIT(S) SUBMITTED ARE POOR QUALITY
- ☐ OTHER: ______________________

IMAGES ARE BEST AVAILABLE COPY.
As rescanning these documents will not correct the image problems checked, please do not report these problems to the IFW Image Problem Mailbox.